Theoretical Computer Science 438 (2012) 55–61

Contents lists available at SciVerse ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

New results on single-machine scheduling with past-sequence-dependent delivery times

Ming Liu [a,*], Feifeng Zheng [b], Chengbin Chu [a,c], Yinfeng Xu [b]

[a] *School of Economics & Management, Tongji University, Shanghai, 200092, PR China*

[b] *School of Management, Xi'an Jiaotong University, Shannxi, 710049, PR China*

[c] *Laboratoire Génie Industriel, Ecole Centrale Paris, Grande Voie des Vignes, 92295 Châtenay-Malabry Cedex, France*

ARTICLE INFO

Article history:
Received 3 November 2011
Received in revised form 15 February 2012
Accepted 7 March 2012
Communicated by D.-Z. Du

Keywords:
Scheduling
Past-sequence-dependent delivery times
Single-machine

ABSTRACT

Scheduling with past-sequence-dependent (*psd*) delivery times is motivated by questions that arise in the electronic manufacturing industry: an electronic component may be exposed to certain a electromagnetic field while waiting for processing and is required to neutralize the effect of electromagnetism. The time spent on the neutralization process has been modeled as *psd* delivery time in the literature. In this paper, we consider single-machine scheduling problems with *psd* delivery times. We respectively derive polynomial algorithms for the following objective functions: the minimization of the total weighted completion time, the total weighted discounted completion time, the total absolute differences in completion times and the sum of earliness, tardiness and common due date penalty. At last, for the criteria of minimization the total weighted tardiness, we propose a polynomial algorithm to optimally solve the problem under a certain condition.

1. Introduction

In many industries, the manufacturing environment has a great influence on the processing times of jobs. A growing body of evidence shows that the manufacturing environment or waiting time may have an adverse effect on the total processing time of a job before delivery to the customer. In some situations, the waiting time-induced adverse effect does not impede the job's suitability to be processed. Moreover, it can be eliminated after the main processing of the job with extra time caused. Such an extra time for eliminating adverse effects between the main processing and the delivery of a job is viewed as a *post-sequence-dependent* (*psd*) delivery time. We mention that the treatment of the adverse effect for a job does not occupy any machine and has no relation to the schedule of the job's main processing. For analytical convenience, it is generally assumed that the *psd* delivery time of a job is proportional to the job's waiting time. One case is that the *psd* delivery time is a simple linear function of the waiting time.

One application of scheduling with *psd* delivery time is in the electronic manufacturing industry. An electronic component may be exposed to certain electromagnetic and/or radioactive fields while waiting in the machine's pre-processing area and regulatory authorities require the component to be "treated" (e.g., in a chemical solution capable of removing/neutralizing certain effects of electromagnetic/radioactive fields) for an amount of time proportional to the job's exposure time to these fields. The treatment is performed immediately after the component has been processed on the machine to ensure a guaranteed delivery to the customer [4]. Such a post-processing operation is usually called the job "tail" or the job "delivery time". Unlike the traditional assumption of a job-specific constant delivery time in the scheduling

* Corresponding author. Tel.: +86 013659234252.
E-mail address: minyivg@gmail.com (M. Liu).

0304-3975/$ – see front matter
doi:10.1016/j.tcs.2012.03.009

literature [5], we assume, as in [4] that the *psd* delivery time of a job is proportional to its waiting time in the manufacturing system.

It has been widely studied in recent decades for one scheduling scenario such that the waiting time of a job has an adverse effect on the job's main processing. The scenario with a waiting time-induced adverse effect is mainly partitioned into three subcategories. In the first category, each job has a so-called *deteriorating processing time* such that the processing time of a job increases in its waiting time. The concept of *deterioration* is introduced by Browne and Yechiali [1] and it describes a kind of adverse effect of waiting time. The second category originated from [3], in which the adverse effect must be removed prior to the main processing of the job by performing a setup operation. The authors introduced the concept of *psd setup time* to describe the adverse effect. The third and newest category is from [4], in which the waiting time-induced adverse effect does not impede the suitability to be processed by one machine and the adverse effect shall be removed prior to delivering the job to the customer. They established a model to incorporate the adverse effect of waiting into a post-processing operation by introducing the concept of *psd delivery times*. The treatment of the adverse effect of each job, i.e, the job's *psd* delivery time, must follow immediately the main processing of the job. In this paper we focus on this model.

Koulamas and Kyparisis [4] assumed that the *psd* delivery time of a job is proportional to the job's waiting time, unlike the traditional assumption of a job-specific constant delivery time in the scheduling literature [5]. Koulamas and Kyparisis [4] presented several results on a single-machine scheduling problem. For the makespan minimization problem, they showed that problem $1|q_{psd}|C_{max}$ can be solved in $O(n)$ time by arranging the largest job at last. Further they showed that $1|q_{psd}|L_{max}$, $1|q_{psd}|T_{max}$ and $1|q_{psd}|\sum U_j$ problems can be reduced to the corresponding problems without *psd* delivery times by appropriate transformations for the processing time and the due date of a job which can be solved polynomially.

To the best of our knowledge, for single-machine scheduling problems with *psd* delivery times, no objective functions except the ones mentioned above have been investigated. In this paper, we further deal with single-machine scheduling problems with *psd* delivery times. The remaining part of this paper is organized as follows. In Section 2, we formulate the models. In Section 3, we consider several single-machine scheduling problems and derive polynomial time algorithms. The last section summarizes the contribution.

2. Problem description and notation

Consider a single machine to process a set of n jobs which are all available at time zero. Given a non-preemptive processing schedule, let p_j, w_j and d_j denote the processing time and the weight and the due date of job J_j, respectively. We use $J_{[j]}$ to indicate the job occupying the j-th position in a schedule. Denote by $S_{[j]}$ the starting time of job $J_{[j]}$ in a schedule. In the environment with *psd* delivery times, the processing of $J_{[j]}$ must be followed immediately by its *psd* delivery time $q_{[j]}$. As in [4], it is assumed that $q_{[j]}$ is proportional to the waiting time or starting time of job $J_{[j]}$. Consequently, $q_{[j]}$ is formulated as

$$q_{[j]} = \gamma S_{[j]}, \quad j = 1, \ldots, n, \tag{1}$$

where $\gamma \geq 0$ is a normalizing constant. Observe that in a single machine environment where the machine is always available for processing and all the n jobs are available at time zero, the starting time $S_{[j]}$ in a schedule without (redundant) idle time satisfies

$$S_{[1]} = 0,$$
$$S_{[j]} = \sum_{i=1}^{j-1} p_{[i]}, \quad j = 2, \ldots, n.$$

Accordingly,

$$S_{[j]} = \sum_{i=1}^{j-1} p_{[i]}, \quad j = 1, \ldots, n, \tag{2}$$

where we define $\sum_{i=1}^{0} p_{[i]} := 0$.

Let $C_{[j]}$ denote the completion time of job $J_{[j]}$ in a schedule (i.e, the completion time of the processing of $J_{[j]}$ on the machine plus the job's *psd* delivery time). Therefore,

$$\begin{aligned} C_{[j]} &= S_{[j]} + p_{[j]} + q_{[j]} = (1+\gamma)S_{[j]} + p_{[j]} \\ &= (1+\gamma)\sum_{i=1}^{j-1} p_{[i]} + p_{[j]}, \quad j = 1, \ldots, n. \end{aligned} \tag{3}$$

For convenience, we denote the *psd* delivery times given in equation (1) by q_{psd}. Let E_j and T_j respectively denote the earliness and tardiness of job J_j, i.e., $E_j = \max\{0, d_j - C_j\}$ and $T_j = \max\{0, C_j - d_j\}$. In the paper, we consider the minimization of the following objective functions: the makespan $C_{max} = \max\{C_j\}$, the total weighted completion time $\sum w_j C_j$, the total weighted discounted completion time $\sum w_j(1 - e^{-rC_j})$ (where r is a constant number), the total absolute differences in completion times $TADC = \sum_{i=1}^{n}\sum_{j=i}^{n} |C_i - C_j|$, the sum of earliness, tardiness and common due date penalty $ETCP = \sum_{j=1}^{n}(P_1 d + P_2 E_j + P_3 T_j)$ (where P_1, P_2, P_3 are the unit due date penalty, the unit earliness penalty and the unit tardiness penalty), and the total weighted tardiness $\sum w_j T_j$ respectively. Thus, using the three-field notation [2], the corresponding scheduling problems are denoted by $1|q_{psd}|C_{max}$, $1|q_{psd}|\sum C_j$, $1|q_{psd}|\sum w_j C_j$, $1|q_{psd}|\sum w_j(1-e^{-rC_j})$, $1|q_{psd}|TADC$, $1|q_{psd}|ETCP$ and $1|q_{psd}|\sum w_j T_j$, respectively.

3. Single-machine scheduling with *psd* delivery times

In this section, we present some results on scheduling on single machine with the consideration of *psd* delivery times. First, we introduce a useful lemma.

Lemma 1. *Let there be two sequences of numbers x_i and y_i. In addition, the two sequences are of the same length. The sum $\sum_i x_i y_i$ of products of the corresponding elements is the least if the sequences are monotonic in the opposite sense.*

Proof. See p. 261 in [6]. □

3.1. Total weighted completion time

We consider the single-machine scheduling problem to minimize total weighted completion time with *psd* delivery times. Denote the model as $1|q_{psd}|\sum w_j C_j$. We show that Shortest Weighted Processing Time (SWPT) rule is optimal for the problem.

Theorem 1. *For $1|q_{psd}|\sum w_j C_j$, SWPT rule is optimal.*

Proof. We begin this proof with the expression of total weighted completion time. By formula (3),

$$\begin{aligned}\sum_{j=1}^{n} w_{[j]}C_{[j]} &= \sum_{j=1}^{n} w_{[j]}\left((1+\gamma)\sum_{i=1}^{j} p_{[j]} - \gamma p_{[j]}\right)\\ &= (1+\gamma)\sum_{j=1}^{n} w_{[j]}\sum_{i=1}^{j} p_{[j]} - \gamma\sum_{j}^{n} w_{[j]}p_{[j]}. \end{aligned} \tag{4}$$

On the right-hand side of the above second equation, the first item $(1+\gamma)\sum_{j=1}^{n} w_{[j]}\sum_{i=1}^{j} p_{[j]}$ is minimized by the SWPT rule, and the second item $\gamma\sum_{j=1}^{n} w_{[j]}p_{[j]}$ is a constant which has no relation with the processing sequence. This completes the proof. □

Remark 1. For the $1|q_{psd}|\sum C_j$ problem, the SPT rule is optimal.

For an alternative proof of this corollary, readers can be referred to [4].

3.2. Total discounted weighted completion time

Consider the objective of total weighted discounted completion time $\sum w_j(1-e^{-rC_j})$, where r $(0 < r < 1)$ is the discount factor. This is a more general cost function than total weighted completion time. The costs are discounted at a rate $r \in (0, 1)$ per unit time. That is, if job J_j is not completed by time t, an additional cost $w_j r e^{-rt}dt$ is incurred over period $[t, t+dt]$. (Note that $\int_0^{C_j} w_j r e^{-rt}dt = w_j(1-e^{-rC_j})$ where C_j is a variable.) The value of r is usually less than 0.1.

The problem $1|q_{psd}|\sum w_j(1-e^{-rC_j})$ gives rise to a different priority rule, i.e., scheduling jobs in non-increasing order of $\frac{w_j e^{-rp_j}}{1-e^{-r(1+\gamma)p_j}}$. This rule is referred to as the Modified Weighted Discounted Shortest Processing Time (MWDSPT) rule.

Theorem 2. *The MWDSPT rule is optimal for $1|q_{psd}|\sum w_j(1-e^{-rC_j})$.*

Proof. By contradiction. Assume otherwise that there exists another optimal schedule S, in which job J_j immediately precedes job J_k while

$$\frac{w_j e^{-rp_j}}{1-e^{-r(1+\gamma)p_j}} > \frac{w_k e^{-rp_k}}{1-e^{-r(1+\gamma)p_k}}.$$

Let t be the time at which job J_j starts its processing in S. An adjacent pairwise interchange between the two jobs results in a new schedule S'. We observe that for each of the other jobs except J_j and J_k, its start time and end time in the two sequences are the same. So, the only difference between S and S' related to the objective value is due to jobs J_j and J_k. Together with formula (3), the total contribution of the two jobs to the objective under S is

$$\begin{aligned}&w_j(1-e^{-r(t+p_j+q_j)}) + w_k(1-e^{-r(t+p_j+p_k+q_k)})\\ &\quad= w_j(1-e^{-r(t+p_j+\gamma t)}) + w_k(1-e^{-r(t+p_j+p_k+\gamma(t+p_j))}).\end{aligned} \tag{5}$$

The total contribution of jobs J_k and J_j to the objective under S' is obtained by interchanging the j and k in formula (5). By algebraic calculation it can be shown that the objective value under S' is less than that under S, implying a contradiction. This completes the proof. □

3.3. Total absolute variation in the job completion times

We consider a scheduling problem with the objective to minimize the total absolute variation in the job completion times (TADC). This scheduling measure was first considered by Kanet [7]. The TADC of the $1|q_{psd}|TADC$ scheduling problem can be computed as follows:

$$\begin{aligned} TADC &= \sum_{r=1}^{n}\sum_{j=r}^{n}|C_{[r]}-C_{[j]}| \\ &= \sum_{r=1}^{n}(2r-1-n)C_{[r]} \\ &= \sum_{r=1}^{n}(2r-1-n)\left[(1+\gamma)\sum_{i=1}^{r-1}p_{[i]}+p_{[r]}\right] \\ &= \sum_{r=1}^{n}[(r-1)(n-r+1)+r(n-r)\gamma]p_{[r]}. \end{aligned}$$

The above equation can be viewed as the scalar product of two vectors, the

$$w_r = (r-1)(n-r+1)+r(n-r)\gamma$$

and $p_{[r]}$ vectors respectively ($r = 1, \ldots, n$). Based on the above analysis and Lemma 1, the optimal sequence for the $1|q_{psd}|TADC$ problem can be obtained in $O(n\log n)$ time by arranging the elements of the w_r and $p_{[r]}$ vectors in opposite orders.

3.4. ETCP problem

In this subsection, we deal with one machine scheduling problem with *psd* delivery times where all jobs have a common due date d. The objective is to determine the optimal value of this due date and an optimal sequence to minimize a total penalty function. This penalty function, $\sum_{r=1}^{n}(P_1d+P_2E_r+P_3T_r)$ (where P_1, P_2, P_3 are the unit due date penalty, the unit earliness penalty and the unit tardiness penalty), is based on the due date value and on the earliness or the tardiness of each job in the selected sequence. If there is no *psd* delivery times (i.e., $\gamma = 0$), the problem is reduced to the $1||\sum_{r=1}^{n}(P_1d+P_2E_r+P_3T_r)$ problem which is addressed in [8]. They provided several useful results of the $1||\sum_{r=1}^{n}(P_1d+P_2E_r+P_3T_r)$ problem.

Observation 1 *([8]). If $P_1 \geq P_3$, the optimal due date $d^* = 0$ and SPT rule is optimal.*

This result holds for the $1|q_{psd}|ETCP$ problem since the objective function does not change. We assume $P_1 < P_3$ throughout the remainder of this subsection, since otherwise the problem is trivial.

Lemma 2 *([8]). It is optimal to assign the due date at the completion time of the Kth job, where K is the smallest integer greater than or equal to $n(P_3-P_1)/(P_2+P_3)$.*

It can be verified that the above lemma holds for the $1|q_{psd}|ETCP$ problem. (See the Proofs of Lemma 1 and 2 in [8]).

It is clear that for any sequence, exactly K jobs will be nontardy ($K = 0$ if $P_1 \geq P_3$). Substituting $d = C_{[K]}$, the total penalty or objective function value is equal to

$$\begin{aligned} ETCP &= \sum_{r=1}^{n}(P_1d+P_2E_{[r]}+P_3T_{[r]}) \\ &= \sum_{r=1}^{n}(P_1C_{[K]}+P_2\max\{0, C_{[K]}-C_{[r]}\}+P_3\max\{0, C_{[r]}-C_{[K]}\}). \end{aligned}$$

By equations (3), we get,

$$\begin{aligned} ETCP &= \sum_{r=1}^{n}(P_1C_{[K]}+P_2\max\{0, C_{[K]}-C_{[r]}\}+P_3\max\{0, C_{[r]}-C_{[K]}\}) \\ &= \sum_{r=1}^{K}(P_1+P_2)C_{[K]}-\sum_{r=1}^{k}P_2C_{[r]}+\sum_{r=K+1}^{n}(P_1-P_3)C_{[K]}+\sum_{r=K+1}^{n}P_3C_{[r]} \\ &= \left\{(P_1+P_2)(1+\gamma)K\sum_{j=1}^{K-1}p_{[j]}+(P_1+P_2)Kp_{[K]}\right\}-\left\{\sum_{j=1}^{K-1}P_2[1+(1+\gamma)(K-j)]p_{[j]}+P_2p_{[K]}\right\} \end{aligned}$$

$$
\begin{aligned}
&+\left\{(P_1-P_3)(n-K)(1+\gamma)\sum_{j=1}^{K-1}p_{[j]}+(P_1-P_3)(n-K)p_{[K]}\right\}\\
&+\left\{\sum_{j=1}^{K-1}P_3(1+\gamma)(n-K)p_{[j]}+P_3(1+\gamma)(n-K)p_{[K]}+\sum_{j=K+1}^{n}P_3[1+(1+\gamma)(n-j)]p_{[j]}\right\}\\
=&\sum_{j=1}^{K-1}\{P_1(1+\gamma)n+P_2[(1+\gamma)j-1]\}p_{[j]}\\
&+\{nP_1+(K-1)P_2+P_3\gamma(n-K)\}p_{[K]}+\sum_{j=K+1}^{n}P_3[1+(1+\gamma)(n-j)]p_{[j]}\\
=&\sum_{r=1}^{n}w_r p_{[r]},
\end{aligned}
$$

where

$$
w_r=\begin{cases}P_1(1+\gamma)n+P_2[(1+\gamma)j-1], & j<K;\\ nP_1+(K-1)P_2+P_3\gamma(n-K), & j=K;\\ P_3[1+(1+\gamma)(n-j)], & j>K.\end{cases}
$$

Based on the above analysis and Lemma 1, the following algorithm with the time complexity $O(n\log n)$ is provided to optimally solve the $1|q_{psd}|ETCP$ problem.

Algorithm 1.

Step 1: Assign the optimal due date at the completion time of the Kth job, where

$$
K=\left\lceil\frac{n(P_3-P_1)}{P_2+P_3}\right\rceil.
$$

Step 2: For $r=1,\ldots,n$, calculate

$$
w_r=\begin{cases}P_1(1+\gamma)n+P_2[(1+\gamma)j-1], & j<K;\\ nP_1+(K-1)P_2+P_3\gamma(n-K), & j=K;\\ P_3[1+(1+\gamma)(n-j)], & j>K.\end{cases}
$$

Step 3: Assign the jobs in the following way: the job with the longest normal processing time to the position with the smallest value of w_r, the job with the second longest normal processing time to the position with the second smallest value of w_r, etc.

3.5. Total weighted tardiness

It is well known that problem $1||\sum w_jT_j$ is NP-hard in the strong sense, which is a special case of $1|q_{psd}|\sum w_jT_j$. This implies that $1|q_{psd}|\sum w_jT_j$ is also NP-hard in the strong sense. In this subsection, the main focus is on a polynomially solvable case. Given a job instance, if for any two jobs J_j, J_k with $p_j\le p_k$, we have $d_j\le d_k$ and $w_j\ge w_k$, then we say the job instance has *agreeable* due dates and *agreeable* weights.

Theorem 3. *If a job instance has agreeable due dates and agreeable weights, then the Shortest Processing Time (SPT) rule is optimal for* $1|q_{psd}|\sum w_jT_j$.

Proof. Suppose otherwise in an optimal schedule S, there exist two jobs J_j and J_k with $p_j\le p_k$ such that J_k is scheduled before J_j. By agreeable due dates and weights, we have $d_j\le d_k$ and $w_j\ge w_k$. Performing a *Position Interchange* on jobs J_k and J_j in S while keeping the processing order of all the other jobs unchanged, we obtain a new schedule S'. It suffices to prove that the total weighted tardiness under S' is no more than that of S.

Let A and $P(A)$ be the set of jobs between J_j and J_k and its total processing time in S. Let t be the time at which job J_k starts its processing under S. J_k is immediately followed by A and then J_j. Under the new schedule S', however, J_j will start its processing at time t, and it is immediately followed by A and then J_k. Notice that the total weighted tardiness of all the jobs except J_j and J_k under S' is less than or equal to that under S due to $p_j\le p_k$. The rest of the proof is to show that the total weighted tardiness of J_j and J_k under S' is no more than that under S.

The total weighted tardiness of J_j and J_k under S is

$$
w_kT_k+w_jT_j=w_k\max\{t+p_k+\gamma t-d_k,0\}+w_j\max\{t+p_k+P(A)+p_j+\gamma(t+p_k+P(A))-d_j,0\}.
$$

While under S', the total contribution of J_j and J_k is

$$
w_jT_j'+w_kT_k'=w_j\max\{t+p_j+\gamma t-d_j,0\}+w_k\max\{t+p_j+P(A)+p_k+\gamma(t+p_j+P(A))-d_k,0\}.
$$

Define $\Delta = (w_jT'_j + w_kT'_k) - (w_kT_k + w_jT_j)$. Below we divide the discussion on Δ into two cases, and show that $\Delta \le 0$ holds in both cases.

Case 1: $t + p_k + \gamma t - d_k \ge 0$.

Case 1.1: $t + p_j + \gamma t - d_j \ge 0$.

$$\begin{aligned}\Delta &= [w_j(t + p_j + \gamma t - d_j) + w_k(t + p_j + P(A) + p_k + \gamma(t + p_j + P(A)) - d_k)]\\ &\quad - [w_k(t + p_k + \gamma t - d_k) + w_j(t + p_k + P(A) + p_j + \gamma(t + p_k + P(A)) - d_j)]\\ &= (1 + \gamma)[w_k(p_j + P(A)) - w_j(p_k + P(A))]\\ &\le 0\end{aligned}$$

where the above inequality is due to $w_j \ge w_k$ and $p_j \le p_k$.

Case 1.2: $t + p_j + \gamma t - d_j < 0$.

$$\begin{aligned}\Delta &= w_k[t + p_j + P(A) + p_k + \gamma(t + p_j + P(A)) - d_k] - [w_k(t + p_k + \gamma t - d_k)\\ &\quad + w_j(t + p_k + P(A) + p_j + \gamma(t + p_k + P(A)) - d_j)]\\ &= w_k(1 + \gamma)(p_j + P(A)) - w_j(t + p_k + P(A) + p_j + \gamma(t + p_k + P(A)) - d_j)\\ &\le w_k[(1 + \gamma)(p_j + P(A)) - (t + p_k + P(A) + p_j + \gamma(t + p_k + P(A)) - d_j)]\\ &= w_k[\gamma(p_j - p_k) - (t + p_k + \gamma t - d_j)]\\ &\le -w_k(t + p_k + \gamma t - d_j)\\ &\le -w_k(t + p_k + \gamma t - d_k)\\ &\le 0\end{aligned}$$

where the first inequality is due to $w_j \ge w_k$ and $t + p_k + P(A) + p_j + \gamma(t + p_k + P(A)) - d_j \ge t + p_k + \gamma t - d_k \ge 0$ by the condition of Case 1, the second and third inequalities are due to $p_j \le p_k$ and $d_j \le d_k$ respectively, and the last inequality is due to case condition $t + p_k + \gamma t - d_k \ge 0$.

Case 2: $t + p_k + \gamma t - d_k < 0$.

Case 2.1: $t + p_j + P(A) + p_k + \gamma(t + p_j + P(A)) - d_k \ge 0$.

Case 2.1.1: $t + p_j + \gamma t - d_j \ge 0$.

$$\begin{aligned}\Delta &= [w_j(t + p_j + \gamma t - d_j) + w_k(t + p_j + P(A) + p_k + \gamma(t + p_j + P(A)) - d_k)]\\ &\quad - w_j(t + p_k + P(A) + p_j + \gamma(t + p_k + P(A)) - d_j)\\ &= w_k(t + p_j + P(A) + p_k + \gamma(t + p_j + P(A)) - d_k)) - w_j(1 + \gamma)(p_k + P(A))\\ &< w_k(1 + \gamma)(p_j + P(A)) - w_j(1 + \gamma)(p_k + P(A))\\ &\le 0\end{aligned}$$

where the first inequality is due to case condition $t + p_k + \gamma t - d_k < 0$, and the second inequality is due to $w_j \ge w_k$ and $p_j \le p_k$.

Case 2.1.2: $t + p_j + \gamma t - d_j < 0$.

$$\begin{aligned}\Delta &= w_k(t + p_j + P(A) + p_k + \gamma(t + p_j + P(A)) - d_k) - w_j(t + p_k + P(A) + p_j + \gamma(t + p_k + P(A)) - d_j)\\ &\le w_j[(\gamma p_j - d_k) - (\gamma p_k - d_j)]\\ &= w_j[\gamma(p_j - p_k) + (d_j - d_k)]\\ &\le 0\end{aligned}$$

where the first inequality is due to $w_k \le w_j$ and the second inequality is due to $p_j \le p_k$ and $d_j \le d_k$.

Case 2.2: $t + p_j + P(A) + p_k + \gamma(t + p_j + P(A)) - d_k < 0$.

Case 2.2.1: $t + p_j + \gamma t - d_j \ge 0$.

$$\begin{aligned}\Delta &= w_j(t + p_j + \gamma t - d_j, 0) - w_j[t + p_k + P(A) + p_j + \gamma(t + p_k + P(A)) - d_j]\\ &= -w_j(1 + \gamma)(p_k + P(A))\\ &\le 0\end{aligned}$$

Case 2.2.2: $t + p_j + \gamma t - d_j < 0$.

$$\begin{aligned}\Delta &= 0 - w_j \max\{t + p_k + P(A) + p_j + \gamma(t + p_k + P(A)) - d_j, 0\}\\ &\le 0.\end{aligned}$$

This completes the proof. $\square$

4. Conclusions

This paper presents the new results of scheduling problems with *psd* delivery times. The *psd* delivery time is of great practical value because it depicts one real-life situation which requires a treatment of the adverse effect for a job. The *psd* delivery time model is proposed by Koulamas and Kyparisi [4] and the results on this area are scarce. In this paper, we further study some single-machine scheduling problems and propose several polynomial algorithms.

Acknowledgments

The authors would like to acknowledge the constructive comments by anonymous referees which have improved the presentation of the paper. The first author is supported by the Fundamental Research Funds for the Central Universities (2010KJ035). This research was supported by the National Science Foundation of China under Grants 71101106, 71172189, 71171149, 70832005 and 71090404/71090400.

References

[1] S. Browne, U. Yechiali, Scheduling deteriorating jobs on a single processor, Operations Research 38 (1990) 495–498.
[2] R.M. Graham, E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan, Optimization and approximation in deterministic machine scheduling: a survey, Annals of Discrete Mathematics 5 (1979) 287–326.
[3] C. Koulamas, G.J. Kyparisis, Single-machine scheduling problems with past-sequence-dependent setup times, European Journal of Operational Research 187 (2008) 1045–1049.
[4] C. Koulamas, G.J. Kyparisis, Single-machine scheduling problems with past-sequence-dependent delivery times, International Journal of Production Economics 126 (2010) 264–266.
[5] M. Pinedo, Scheduling: Theory, Algorithms, and Systems, second ed., Prentice Hall, New York, 2005.
[6] G.H. Hardy, J.E. Littlewood, G. Polya, Inequalities, Cambridge University Press, London, 1967.
[7] J.J. Kanet, Minimizing variation of flow time in a single machine systems, Management Science 27 (1981) 1453–1459.
[8] S.S. Panwalkar, M.L. Smith, A. Seidmann, Common due date assignment to minimize total penalty for the one machine scheduling problem, Operations Research 30 (1982) 391–399.